



Taller de Robótica Educativa

Guía de ejercicios – Apuntes

Ejercicios

- Escribir un programa para que el robot se mueva para adelante y atrás una vez.

Tener en cuenta:

- Todo programa debe comenzar con la etiqueta **_start**
- Se debe escribir **una** instrucción por línea
- Usar la instrucción **motor** (ver al dorso)
- Luego de avanzar hacia adelante, el robot debe **prender el led**. Al volver al punto de salida, debe **apagarlo**.

1. Escribir un programa para que el robot se mueva para adelante y atrás infinitas veces.
2. Escribir un programa para que el robot se mueva haciendo un cuadrado infinitas veces.
3. Escribir un programa para que el robot se mueva haciendo **un único** cuadrado, y luego se detenga donde empezó.

¿Qué cambiaría si en vez de un cuadrado quiero hacer un pentágono? ¿Y si quiero un hexágono? ¿Y si quiero un polígono de **N** lados?

4. Opcional:

- Escribir un programa para que el robot haga **un solo** cuadrado usando **registros** y la instrucción **MOV** y **DEC**.
- ¿Qué cambiaría si en vez de un cuadrado quiero hacer un polígono de **N** lados?

Observaciones:

- Una vez escrito el código del programa, se debe guardar el código con un nombre **sin tildes**. Si hacemos clic en **PLAY** y no guardamos antes, la **IDE** nos pedirá guardar el archivo.
- Al hacer clic en **PLAY** aparecerá una ventana donde podemos elegir un **fondo** y una distribución de **cajas** para el simulador. Se debe hacer clic en **simular** sin elegir nada para usar la configuración por **defecto**.
- Cuando estamos en el simulador, se puede volver a la IDE presionando la tecla escape.
- Utilizando “;” (punto y coma) al final de las instrucciones es posible insertar comentarios en el código.
- **¡Tu consulta no molesta! Preguntá por cualquier duda que tengas al resolver los ejercicios**

Anotaciones propias:

Manual de Instrucciones

Introducción a las instrucciones principales

LED cual, nuevo_estado

Prende o apaga un led. Por ahora solo tenemos un led disponible, por lo que el parámetro 'cual' será siempre cero. El parámetro 'nuevo_estado' es **0** para **apagar** o **255** para **prender**.

Ejemplo:

```
LED 0 255    ;prende el LED
LED 0 0      ;apaga el LED
```

MOTOR tiempo, motor_izq, motor_der

Por una cantidad de tiempo prende los motores con la potencia que se indican en motor_izq y motor_der. Las potencias pueden ir de -127 a 127 donde un número negativo indica ir en reversa.

Ejemplos:

MOTOR 10, 127, 127

El robot avanza para adelante 10 unidades de tiempo a máxima potencia

MOTOR 10, 127, 0

Girar a la izquierda usando una rueda como eje de rotación por 10 unidades de tiempo. En este caso, el motor izquierdo tiene máxima potencia y el derecho está apagado.

MOTOR 10, 127, -127

Girar a la izquierda usando el centro del robot como eje de rotación por 10 unidades de tiempo. La diferencia con el caso anterior es que el motor derecho está andando en reversa, haciendo el giro más pronunciado.

MOTOR 10, -127, -127

Ir en reversa.

JMP etiqueta

Se pasa a ejecutar código a partir de la ubicación de **etiqueta**.

Ejemplo:

```
_start:
    LED 0, 255    ;prendo el led
    MOTOR 10, 0, 0;espero 10 unidades de tiempo con los motores apagados
    LED 0, 0      ;apago el led
    JMP _start    ;ejecuto desde _start
```

En este ejemplo, se prende y apaga el led infinitamente.

Las instrucciones MOV y DEC

El siguiente es un ejemplo que usa la instrucción MOV junto con DEC y registros.

```
_start:
    MOV r0, 2          ;muevo 2 a r0 (¡r0 es un registro!)
_mover:
    MOTOR 15, 120, 120 ;avanzar
    DEC r0              ;decremento r0 en uno
    SKIPZ r0            ;si r0 está en cero ignoro la siguiente instrucción
    JMP _mover          ;repetir desde _mover
```

El ejemplo avanza el robot mientras el registro **r0** no tenga el valor cero (lo hará dos veces).